06-16-00 A

TRANSMITTAL FORM FOR FILING PATENT APPLICATION

Attorney
Docket No.: MCL-001XX

WEINGARTEN, SCHURGIN, GAGNEBIN & HAYES LLP
Ten Post Office Square
Boston, Massachusetts 02109
Telephone: (617) 542-2290
Telecopier: (617) 451-0313

Express Mail No: EL418426925US

BOX PATENT APPLICATION
Assistant Commissioner for Patents
Washington, D.C. 20231

Date: June 15, 2000

First Named Inventor or Application Identifier: Brian Stevens

Sir:

Transmitted herewith under 37 CFR § 1.53 for filing is the patent application of:

Inventor: Brian Stevens

Entitled: SECURE REMOTE SERVICING OF A COMPUTER SYSTEM OVER A COMPUTER NETWORK

[X] This is a request for filing a [ ] **continuation** [ ] **divisional** [ ] **continuation in-part** application under §1.53(b) of prior Application No._____, filed _____ entitled:

Enclosed are:

[X] 23 pages of written description, claims and Abstract, inclusive

[X] 5 sheets of [X] informal [ ] formal drawings of Figs. 1-4c (one set)

[X] Oath or Declaration
- [X] Newly executed (original)
- [ ] Copy from prior application (37 CFR 1.63(d)) (for continuation/divisional). The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
- [ ] To be filed later

[X] Cover sheet and Assignment of the invention to: Mission Critical Linux, LLC

[ ] Certified copy of a _____ application (if foreign priority is claimed) with letter claiming priority under Rule 55.

[ ] Information Disclosure Statement with __ citations

[ ] Preliminary amendment is enclosed.

[X] Return receipt postcard

[ ] Other:

WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES LLP
TEL (617) 542-2290
FAX (617) 451-0313

TRANSMITTAL FORM FOR FILING PATENT APPLICATION (CONTINUED)

✓ Attorney
Docket No.: MCL-001XX

[X] Verified statement of Small Entity status (§1.9 and §1.27)

[ ] Verified statement of Small Entity was filed in prior application. Status still proper and desired

[ ] Priority is claimed under 35 USC § 120 as indicated on the attached sheet 4.

[ ] Priority is claimed under 35 USC §119(a)-(d) as indicated on the attached sheet 4.

[X] Priority is claimed under 35 USC §119 (e) as indicated on the attached sheet 4.

[ ] _____ is hereby appointed Associate Attorney by:
Registration No.:

_____
Attorney of Record
Registration No.:

[ ] **Power of Attorney** in the originally-filed application has been granted to one or more of the registered attorneys listed below. The attorneys listed below not previously granted power in the originally-filed application, as well as _____, are hereby given associate power:
Registration No.:

Stanley M. Schurgin, Reg. No. 20,979
Charles L. Gagnebin III, Reg. No. 25,467
Paul J. Hayes, Reg. No. 28,307
Victor B. Lebovici, Reg. No. 30,864
Eugene A. Feher, Reg. No. 33,171
Beverly E. Hjorth, Reg. No. 32,033
Holliday C. Heine, Reg. No. 34,346
Gordon R. Moriarty, Reg. No. 38,973

[ ] Cancel in this application original claims _____ of the prior application before calculating the filing fee.

[ ] Add in this application claims _____ per amendment before calculating fee.

CLAIMS FILED:	MINUS BASE:	EXTRA CLAIMS:	RATE:	BASIC FEE:
				$690.00
Independent	4 - 3	= 1	x $78.00 =	78.00
Total	28 - 20	= 8	x $18.00 =	144.00
[ ] Multiple Dependent Claims (1st presentation)			+ $260.00 =	0
			SUBTOTAL FILING FEE	912.00
Small Entity filing, divide by 2. (Note: verified statement must be attached per §1.9, §1.27, §1.28.)				456.00
			TOTAL Filing Fee	$456.00

WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES LLP
TEL (617) 542-2290
FAX (617) 451-0313

Attorney Docket No.: MCL-001XX

TRANSMITTAL FOR FILING PATENT APPLICATION (CONTINUED)

[X] The filing fee has been calculated above; a check in the amount of $456.00 is enclosed.

[ ] The filing fee will be submitted at a later date.

[X] In the event a Petition for Extension of Time under 37 CFR §1.17 is required by this paper and not otherwise provided, such Petition is hereby made and authorization is provided herewith to charge Deposit Account No. 23-0804 for the cost of such extension.

[X] The Commissioner is hereby authorized to charge payment of any additional filing fees under 37 CFR §1.16 associated with this communication or credit any overpayment to Deposit Account No. 23-0804.

Address all future communications to:

WEINGARTEN, SCHURGIN, GAGNEBIN & HAYES LLP
Ten Post Office Square
Boston, Massachusetts 02109
Telephone: (617) 542-2290
Telecopier: (617) 451-0313

Attorney of Record: Gordon R. Moriarty
Registration No. 38,973

Attorney Docket No.: MCL-001XX

TRANSMITTAL FOR FILING PATENT APPLICATION (CONTINUED)

[ ] Priority is claimed under 35 USC § 120 of prior Application(s) No. _____, filed_____, entitled:

[ ] The above-identified application(s) is/are assigned of record to:

[ ] Priority is claimed under 35 USC § 119 (a)-(d) of the following application(s).

(Application Number)	(Country)	(Filing Date)

[ ] The above-identified application(s) is/are assigned of record to:

[X] Priority is claimed under 35 USC § 119 (e) of the following provisional application(s).

(Application Number)	(Filing Date)
60/160,985	October 22, 1999

[X] The above-identified provisional application(s) is/are assigned of record to: Mission Critical Linux, LLC

[ ] The claim of small entity status in the above-identified provisional application(s) is made in this application and a copy of the small entity form(s) from the provisional application(s) is/are enclosed.

SUBMIT IN TRIPLICATE

225615

WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES LLP
TEL (617) 542-2290
FAX (617) 451-0313

2/98 FORM 10 Sheet 1 of 2

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT: Brian Stevens

ATTORNEY DOCKET NO.: MCL-001XX

APPLICATION NO.:

EXAMINER:

FILED:

GROUP NO.:

PATENT NO.:

ISSUED:

ENTITLED: SECURE REMOTE SERVICING OF A COMPUTER SYSTEM OVER A COMPUTER NETWORK

VERIFIED STATEMENT AS SMALL ENTITY

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

THE UNDERSIGNED DECLARE(S):

Exclusive rights in the above-identified invention reside in the "small entity(ies)" defined and named below or in a Verified Statement as Small Entity filed by other such small entity(ies), and "small entity" fees are appropriate. Qualification as a small entity is based upon the appropriately checked statements below:

[ ] INDEPENDENT INVENTOR(S)

The below-signing independent inventor(s) has (have) not assigned, granted, conveyed or licensed, and is (are) under no obligation under contract or law to assign, grant, convey or license any rights in the invention to any person who could not likewise be classified as an independent inventor under 37 CFR 1.9(c) if that person had made the invention, or to any concern which would not qualify as a small business concern under 37 CFR 1.9(d) or a nonprofit organization under 37 CFR 1.9(e).

[X] SMALL BUSINESS CONCERN

The below-identified small business concern qualifies as a small business as defined in 13 CFR 121.1301 through 121.1305, and reproduced in 37 CFR 1.9(d), for purposes of paying reduced fees under Section 41(a) and (b) of Title 35, in that the number of employees, including those of its affiliates, which does not exceed 500 persons, and it has not assigned, granted, conveyed or licensed, and is under no obligation under contract or law to assign, grant, convey or license, any rights in the invention to any person who could not be classified as an independent inventor under 37 CFR 1.9(c) if that person had made the invention, or to any concern which would not qualify as a small business concern under 37 CFR 1.9(d) or a nonprofit organization under 37 CFR 1.9(e).

Express Mail No.

EL418426925US

2/98 FORM 10

Attorney Docket No.: MCL-001XX

Concerns are affiliates of each other when, either directly or indirectly, one concern controls or has the power to control the other, or a third party controls or has the power to control both. The number of employees of the business concern is the average over the fiscal year of the persons employed during each of the pay periods of the fiscal year. Employees are those persons employed on a full-time, part-time or temporary basis during the previous fiscal year of the concern.

[ ] NONPROFIT ORGANIZATION (Check additional applicable box.)

The below-identified nonprofit organization qualifies as a small entity under 37 CFR 1.9(e) in that it constitutes:

1. [ ] a university or other institution of higher education located in any country; or

2. [ ] an organization of the type described in Section 501(c)(3) of the Internal Revenue Code of 1954 (26 USC 501(c)(3)) and exempt from taxation under Section 501(a) of the Internal Revenue Code (26 USC 501(a)); or

3. [ ] any nonprofit scientific or educational organization qualified under a nonprofit organization statute of a state of the United States (35 USC 201(i)); or

4. [ ] any nonprofit organization located in a foreign country which would qualify as a nonprofit organization under paragraphs (e)(2) or (3) of Rule 1.9 if it were located in the United States.

The undersigned acknowledge(s) the duty to file, in this application or patent, notification of any change in status resulting in loss of entitlement to small entity status prior to paying, or at the time of paying, the earliest of the issue fee or any maintenance fee due after the date on which status as a small entity is no longer appropriate (37 CFR 1.28(b)).

The below-signing individual(s) hereby declare(s) that (he, she, they) are authorized to execute this statement on behalf of the small entity.

Name of Small Entity: (Small Business) Mission Critical Linux, LLC	
Address of Small Entity: (Street, City, State or Country, Zip Code) 100 Foot of John Street; 3rd Floor, Lowell, Massachusetts 01852	
Name of Person Signing: (Small Business) Moiz Kohari	
Title of Person Signing: President and Chief Executive Officer	
Signature: (Please sign and date in permanent ink.) **X**	**Date signed:** **X** 3/29/2000

223123

TITLE OF THE INVENTION

SECURE REMOTE SERVICING OF A COMPUTER SYSTEM OVER A COMPUTER NETWORK

CROSS REFERENCE TO RELATED APPLICATIONS

This application claims priority of U.S. Provisional Patent Application No. 60/160,985, filed October 22, 1999, the disclosure of which is hereby incorporated by reference.

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

N/A

BACKGROUND OF THE INVENTION

This invention relates to the servicing of computer systems, and more specifically to the secure servicing of remote computer systems and network appliances.

The servicing of computer systems encompasses the processes and methods by which the proper operation and maintenance of computer systems are ensured. Servicing can be used to detect and correct problems prior to serious failures, or to restore computer system operation when the proper functioning of the system has been compromised.

With an ever expanding use of computer systems in the commercial sector, many businesses find outsourcing the servicing of their computing investments a cost effective alternative to maintaining and managing an internal support staff. However, implicit in the contracting of third party support has been the cost of support personnel travelling to the contracting party's site to perform such servicing.

ATTORNEY DOCKET NO. MCL-001XX
WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES, LLP
TEL. (617) 542-2290
FAX. (617) 451-0313

Further, there is often a delay in the provision of such service, particularly when the contracting party is located in a location remote from significant centers of commerce. While it may be possible for a contracting party to request expedited on-site support when needed, such a request typically comes at an increased cost.

The wide scale use of computer networks such as the Internet has not yet been leveraged as an effective tool for the servicing of computer systems. Today, the Internet's role is relegated to being the conduit by which problem reports are entered and tracked by system administrators. In most cases, such problem reports provide only the symptoms or consequences of a problem; service professionals must still obtain additional information in order to provide an effective resolution. Access to this additional information typically occurs in one of three ways: over the phone with remote service professionals instructing system administrators with commands to run various directives and to report verbally on their outcome; through email exchanges between the service professional and the system administrator; or by on-site staffing or visitation by service professionals. The former two have historically provided a slow problem resolution time due to the need for information to pass through an intermediary, the system administrator. The information relayed to the service personnel may also be incomplete or improperly characterized. The latter approach clearly enables rapid resolution, but can be significantly more expensive than the other methods.

It would therefore be preferable to enable a system for supporting secure computer networks that combines the beneficial aspects of these prior art approaches.

Specifically, such a support system would preferably have the following three characteristics.

First, the system must be interactive, such that the service professional is capable of directly interrogating or commanding the computer system to be maintained and of receiving a direct, substantially immediate response. The service professional should have the ability to download patches and make changes to the target system in order to restore compromised function as rapidly as possible.

Second, this interactive access to the target system must be capable of being provided remotely in order to obviate the need for as-needed or permanent on-site support. The service professional must have the ability to access the system being serviced from any location having access to the Internet or other appropriately configured data network.

Third, the facility for providing service personnel with remote access must be secure. Only authorized service personnel should have access to the target computer system. Further, all data exchanges between the service personnel and the target system should be encrypted to prevent electronic eavesdropping or "snooping" by third parties. Encryption also serves to frustrate attempts by third parties to inject false directives to the target system or to submit false data to the service personnel.

The Internet provides the communications vehicle by which businesses all over the world are connected. Layered protocols such as Hypertext Transport Protocol (HTTP) support interactive exchanges over the Internet. It is necessary for businesses to tightly control which, if any, parts of their internal computer networks are accessible to computer users outside such internal networks. This is often accomplished through the use of firewall technology

which segment an enterprise's networks such that internal networks are not accessible to unauthorized personnel including users of other networks such as the Internet. To this end, firewalls examine data packet identifiers in deciding which are allowed to pass the boundary between internal and external networks.

However, necessary security provisions including firewalls represent an obstacle to realizing a secure, remote network support system. For instance, an attempt to send a request other than a mail message to a firewall-protected network will normally fail. The only systems which are accessible by external access are corporate web servers which are often resident outside the firewall.

The most common remote management solution in use at present is Secure Shell (SSH) which allows encrypted, remote login over the Internet. The system allowing remote login must manage all access control; improper configuration of such a system could expose the protected systems to a security risk. In addition, firewalls between the Internet and a system to be supported must be configured to allow a port specific to SSH to be passed through, which some administrators are reluctant to do.

BRIEF SUMMARY OF THE INVENTION

The presently disclosed remote servicing of a secure computer system employs an intermediate network entity accessible to both a remote service provider and to an agent running on the target computer system to be serviced. Such servicing may be referred to as Secure Servicing Technology (SST).

A service professional's computer runs a Service Manager (SM) software module, while the system being managed

or serviced runs a Secure Service Agent (SSA) software module.

A Secure Service Intermediary (SSI) software module runs on the intermediate network entity such as a computer system accessible to both the SM and the SSA. This mutually accessible system may be located outside firewalls protecting the system to be maintained or inside such firewalls configured to allow selected access. In addition, the mutually accessible system is configured with all of the safeguards of computer systems currently supporting e-commerce, including encrypted connections.

Access to the intermediate network entity is limited to access over secure access protocols, and then only after proper authentication and authorization. The SM, using secure access protocols, connects to the SSI, and after authentication authorizes itself to perform directives on the SSA. A directive may be any command or executable which the SM requests to be executed on the SSA. Once so authorized, the SM passes a directive via standard HTTP to a CGI process spawned by the SSI. The CGI establishes whether a secure connection exists between the SSI and the target SSA. If so, the CGI process passes the directive to the SSI, which in turn passes the directive to the SSA using secure access protocols. The SSA then executes the directive. Meanwhile both the SSI and the CGI process block, or enter an idle state, until a valid response is returned from the SSA to the SSI. The SSI passes the response to the CGI, which in turn provides it to the SM for analysis by the service professional.

This flexible architecture has application to a broad range of networked computer systems requiring interactive,

secure data exchange over a distributed network such as the Internet.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING

Fig. 1 is a block diagram of one embodiment of a remote support system for a secure computer system according to the present invention;

Fig. 2 is a block diagram of one embodiment of software modules which are executed by the secure computer system of Fig. 1;

Fig. 3 is a flow diagram illustrating the establishment of secure communications between a secure service agent running on a computer system to be serviced and a secure service intermediary; and

Fig. 4 is a flow diagram illustrating the establishment of secure communications between a remote service manager and a secure service intermediary, as well as the transmission of service directives from the service manager to the secure service agent via the secure service intermediary and the return of resulting data from the secure service agent to the service manager via the secure service intermediary.

DETAILED DESCRIPTION OF THE INVENTION

The present system 10 for remotely supporting or servicing secure computer systems 14 enables a service professional to maintain a computer system anywhere in the world.

With reference to Figs. 1 and 2, the service professional employs a computer system running a service manager (SM) module 12. In a first embodiment, the SM is an Internet browser 32 such as NETSCAPE NAVIGATOR. The service

professional employs the SM 12 to send service directives to and to receive the results of executed service directives from a secure service intermediary (SSI) 16. The data pathway between the SM 12 and the SSI 16 is the Internet in a first embodiment, with world-wide interconnectivity enabling remote access for a service professional from practically any location. However, in alternative embodiments, dedicated or limited access computer networks are utilized.

In a first embodiment, the SSI 16 is realized by a web server 34 which supports secure sockets layer (SSL), such as APACHE of Apache Micro Peripherals, Inc. SSL is a protocol developed by Netscape Communications Corporation for transmitting private documents or data via the Internet. SSL works by using a private key to encrypt data that is transferred over the SSL connection.

Much of the functionality of the SSI 16 is implemented in common gateway interfaces (CGI) 36. CGI is a standard for external gateway programs which interface with information servers such as World Wide Web servers. A CGI program is also any program designed to accept and return data that conforms to the CGI specification. The program can be written in any programming language, including C, Perl, Java, or Visual Basic. CGI programs are the most common way for web serves to interact dynamically with users.

The SSI 16 is also comprised of a parent intermediary daemon process 38. A daemon process is a software process that runs in the background and performs a specified operation. This parent intermediary daemon process 38 is responsible for conveying directives received from the SM 12 to the SSA 18, as will be described below. In addition, the

parent intermediary daemon process 38 is responsible for conveying results from one or more SSA-executed directives to the appropriate SM 12, as will be discussed in the following.

Service personnel ("the user") can connect to the SSI 16 via the Internet 28 by entering the appropriate universal resource locator (URL) into the SM browser 32. Once connected, the user is requested to enter his or her name and password such that a CGI process 36 of the SSI 16 can authenticate the identity of the user and ensure that the user is an authorized user of the SSI system 16. All network (i.e. Internet) transactions between the SM browser 32 and the SSI web server 34 use SSL encryption for complete protection of passwords and data.

Once authorized, the user selects which secure computer system 14 ("target system") they wish to perform a service management directive on. The user is then presented with a web-based interface that allows them to query and perform actions on the selected system to be supported 14 via a secure service agent (SSA) 18.

The SSA 18 is implemented as a software module capable of executing various commands or executables on the target system(s) to be serviced 14. In a preferred embodiment, the SSA 18 is implemented as a parent agent daemon process 44. When the SSA 18 is initiated, the parent agent daemon process 44 establishes a secure connection to the SSI 16 parent intermediary daemon process 38.

When a directive for execution by one of the target systems 14 has been received over this secure connection, the parent agent daemon process 44 spawns a respective child agent daemon process 46. This child agent daemon process 46 carries out the execution of the directive and returns any

results to the parent agent daemon process 44. These daemon programs 44, 46 are written in the C programming language in a first embodiment.

As the user selects a service or maintenance operation, a CGI program 36 on the SSI 16 determines whether a valid, secure connection has been established between an child intermediary daemon process 40 and the parent agent daemon process 44. If such a connection exists, the SSI CGI process 36 makes a connection to the this child intermediary daemon process 38 and passes the directive received from the SM 12 to this child intermediary daemon process 40. The child intermediary daemon process 40 receives the directive and sends it over the secure connection to the parent agent daemon process 44 on the SSA 18. Having transferred the directive to the SSA 18, both the CGI process 36 and the child intermediary daemon process 40 block, or enter an idle state, pending receipt of results from the execution of the directive by the target system 14.

Meanwhile, if the user issues another directive to the SSI 16, the SSI 16 spawns a new CGI process 36 to receive the directive. The new directive is then transferred by this new CGI process 36 to the child intermediary daemon process 40 which is already in communication with the respective SSA 18 processes. This child intermediary daemon process 40 then forwards the new directive to the parent agent daemon process 44, which in turn spawns a new child agent daemon process 46 for executing the directive on the target system and which returns the results in the same fashion as previously described. Thus, several users and/or directives can be passed to the SSI 16 and on to the SSA 18 at the same time. A single connection between the SSI 16 child intermediary daemon process 40 and the parent agent

daemon process 44 is capable of supporting multiple users and/or simultaneous directives.

Once a parent agent daemon process 44 on an SSA 18 associated with a target system to be maintained 14 is started or initiated and the communications channel with the respective child intermediary daemon 40 process has been established, the parent agent intermediary process 44 blocks pending receipt of a directive to be executed on one of the target systems 14 associated with that SSA 18. If such a directive is received, the parent agent daemon process 44 spawns a respective child agent daemon process 46 which will remain in communication with the parent agent daemon process 44 until all results from the executed directive have been returned to the SSI 16. Once this occurs, the child agent daemon process 46 exits. Each of these daemon processes 38, 40, 4, 46 block, or remain in an idle state, until they are needed to either receive or send data.

As with the SM browser 32, the connection from the SSA 18 to the SSI 16 uses hypertext transfer protocol (HTTP) over SSL. In a first embodiment, the data network between the SSA 18 and the SSI 16 is the Internet 28, though other networks are employed in further embodiments. HTTP, the underlying protocol used by the World Wide Web, defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands. Also as with the SM browser 32, the identity of the SSA 18 is authenticated by the SSI 16 parent intermediary daemon process 38. A firewall 22 separating the SSI 16 and the systems to be serviced 14 is configured in order to allow selective access to the SSI 16 by the SSA 18. Any attempt to transmit directives from external

devices directly to the SSA 18 (other than through the SSI 16 as described above) is prevented.

Once connected to and authenticated by the SSI 16 daemon process at startup, the SSA 18 daemon 44 blocks, and receives directives from and sends results to the SSI 16 daemon process over the valid SSL connection. Once the SSI 16 child intermediary daemon process 40 acknowledges a response from the respective agent process 44, the SSI child intermediary daemon process 40 becomes unblocked. The response from the agent daemon 44 is received by the respective child intermediary daemon process 40, then sent over the valid connection to the SSI CGI process 36, which in turn sends the results to the SM 12 web browser 32 for display to the user.

As noted, all communications between the SM 12, SSI 16 and SSA 18 employ HTTP over SSL. Thus, all data is encrypted end to end, and the SSI is authenticated using digital certificates each time a new SSA 18 makes an initial connection. Additionally, user identity and password information are validated by the SSI 16 with each directive or directive response.

An optional firewall 20 is provided between the SM 12 and the SSI 16. In this embodiment, the firewall 20 is configured to allow selected access by each of the SM 12 to the SSI 16 according to methods known to one skilled in the art.

With reference to Fig. 3, the process by which the SSA 18 receives directives from the SSI 16 is illustrated. For purposes of this illustration, the SSA 18 may be referred to as "the Agent" and the SSI 16 may be referred to as "the Intermediary" or "the web server."

Depending upon the prior state of the system to be managed or serviced 14, the parent agent daemon process 44 of the Agent 18 is initiated (100), and authentication information is provided, either manually or by reference to a secure configuration file (102). Upon initial receipt of contact by the parent agent daemon process 44 (104), the parent intermediary daemon process 38 responds to the Agent 18 with a digital certificate (106). This enables the parent agent daemon process 44 to authenticate the Intermediary 16 (108).

Once authenticated, the Intermediary 16 receives a site-specific password from the Agent 18 (110). This password is stored in a secure or protected database in the Intermediary 16 (112) for later use in authenticating a user attempting to provide the Agent 18 with directives via the Intermediary 16. In one embodiment, the validity of the transferred password has a discrete lifetime. Thus, the password may be stored in the secure database of the Intermediary along with a time-stamp.

The parent intermediary daemon process 38 also spawns a child intermediary daemon process 40 in response to the establishment of a valid connection between the Intermediary 16 and the Agent 18. This child intermediary daemon process 40 will remain in existence as long as the valid connection exists between the Intermediary 16 and the Agent 18. Once spawned, the child intermediary daemon process 40, and the parent agent daemon process 44, block, receive or send (114a, 114b), depending upon whether a directive exists for transfer from the Intermediary 16 to the respective Agent 18 or whether directive results exist for transfer from the respective Agent 18 to the Manager 12 via the Intermediary 16.

With reference to Fig. 4, one process by which a directive is defined by the Manager 12 and is responded to by the Agent 18 is illustrated. First, a service professional, referred to as a "user," opens a web browser 32 using the Manager 12 (200). Once the browser has been initiated, the user provides the browser with the URL of a login page in the Intermediary 16 web server 34 (202), resulting in the browser 32 sending a connect request to the web server 34 of the Intermediary 16 (204).

The web server 34 responds to the connect request by returning a digital certificate to the Manager 12 (206), which the Manager 12 uses to authenticate the Intermediary 16 (208). The web browser 32 spawns a CGI process in the Manager 12 for this purpose. If successful, the Manager 12 requests and receives a login page from the Intermediary 16 web server 34 (210, 212). The user provides his/her account name and password to the Intermediary 16 through this page and an associated CGI process (214), and the web server 34 spawns a CGI process for receiving this information and for authorizing the user via another Manager CGI process (216). Both the Manager 12 and Intermediary 16 have now been authenticated (218).

Next, the user enters a URL into the browser 32 which results in the Intermediary 16 spawning a CGI process which references an Intermediary 16 database (not illustrated) to establish which Agents 18 are active, and of those, which the user is authorized to access (220, 222). A list of user-accessible Agents 18 is provided to the Manager 12 browser 32 for user selection (222, 224), and the user's selection is uploaded to the Intermediary 16 via a CGI process (226).

In response to receipt of the Agent identifier, the Intermediary 16 web server 34 spawns a CGI process that requests from the user the site-specific password previously stored in the protected database of the Intermediary 16 (228, 110, 112). The user enters the password via the web browser 32 and another CGI process (230). Following Intermediary CGI authorization of the user based on the entered password (232), the web server 32 spawns a CGI process for enabling the user to define directives for execution by the Agent (234).

In response to the authorization confirmation by the Intermediary 16, the user enters into the Manager browser 32 a directive that he/she wants to have executed by the Agent 18 (234). This directive request is then uploaded to the Intermediary web server 34 (236), which spawns a CGI process 36 (238) for confirming that a valid SSL connection between the child intermediary daemon process 40 and the parent agent daemon process 44 has been established. If such a valid connection exists, the CGI process 36 passes the directive over a local SSL connection to the child intermediary daemon process 40. The CGI process 36 then begins blocking (238) on the local connection until a valid response is received from the child intermediary daemon process 40. This child intermediary daemon process meanwhile receives the directive and forwards it to the parent agent daemon process 44 (240) associated with the target system to be serviced 14.

Meanwhile, as indicated at the end of Fig. 3, the parent agent daemon process 44 has been blocking over the SSL with the Intermediary 16 while waiting to receive directives (114, 242). The Agent 18 is capable of handling plural directives simultaneously because it spawns a

respective child agent daemon process 46 for controlling each directive to be executed on a respective target system 14 (244). The same child agent daemon process 46 blocks until results from the executed directive are received from the respective target system 14 (246). All responses from executed directives are returned to the Intermediary 16 over the same SSL connection via communication from the child agent daemon process 46 to the parent agent daemon process 44 to the child intermediary daemon process 44 (246).

The intermediary CGI process 36, which had been blocking, receives the response from the parent agent daemon process 44 via the child intermediary daemon process 40 (248) and sends it via standard HTTP to the Manager web browser 32 (250) for display at the user's computer 12 (252). The same CGI process 36 continues to block/receive/send, waiting for further results from the executed directive previously conveyed to the Agent 18 (254). Once the child agent daemon process 46 determines that the directive has been executed by the target system 14 and all responses from this directive have been passed back to the Intermediary 16, the child agent daemon process 46 exits (256). Meanwhile, the parent agent daemon process 44 continues to block (242) while waiting to receive further directives which would again cause the parent agent daemon process 44 to spawn further child agent daemon processes 46.

In the Intermediary 16, the child intermediary daemon process 40 and the CGI process 36 continue to block pending receipt of further results from the previously transferred directive (258). Once the child intermediary daemon process 40 and the CGI process 36 transfer the last results from the executed directive to the Manager browser 32, the CGI process exits while the child intermediary daemon process

continues blocking pending receipt of a new directive to be conveyed to the respective parent agent daemon process 44 (260).

In the browser 32 of the Manager 12, the data from the Intermediary 16 is received and displayed to the user (262). The user then has the option of submitting one or more subsequent directives to the Intermediary for execution by the selected Agent (234).

As noted, the term "directive" can be interpreted as a single command, a string of diagnostic commands, or a command to replace an existing set of executable code with a new set. Other interpretations are also possible.

ATTORNEY DOCKET NO. MCL-001XX
WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES, LLP
TEL. (617) 542-2290
FAX. (617) 451-0313

CLAIMS

1. A remote access system for a secure computer network, comprising:

an agent operative in conjunction with said secure computer network;

an intermediary operative in conjunction with a network entity distinct from said secure computer network; and

a manager for defining a directive to be executed by said agent, wherein

said manager and said intermediary are capable of establishing secure communications therebetween,

said intermediary and said agent are capable of establishing secure communications therebetween, and

said manager is operative to convey said directive to be executed to said intermediary and said agent is operative to receive said directive to be executed from said intermediary prior to executing said directive.

2. The system of claim 1, wherein said agent is operative to execute said directive and to return results of said executed directive to said intermediary.

3. The system of claim 2, wherein said intermediary is operative to forward said results of said executed directive to said manager subsequent to said return of said results to said intermediary by said agent.

4. The system of claim 1, wherein said agent is operative to block pending receipt of said directive to be executed from manager via said intermediary.

ATTORNEY DOCKET NO. MCL-001XX
WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES, LLP
TEL. (617) 542-2290
FAX. (617) 451-0313

5. The system of claim 1, wherein said agent and said intermediary are capable of authenticating each other prior to establishing said secure communications therebetween.

6. The system of claim 1, wherein said manager and said intermediary are capable of authenticating each other prior to establishing said secure communications therebetween.

7. The system of claim 6, wherein said agent is capable of providing a password to said intermediary for subsequent comparison by said intermediary with a password provided by said manager.

8. The system of claim 6, wherein said agent is capable of providing said intermediary with a identifying data, said intermediary capable of providing said identifying data to said manager, and said manager capable of authenticating said intermediary based upon said identifying data.

9. The system of claim 1, wherein said manager is a web browser and said intermediary is a web server.

10. The system of claim 1, further comprising a first data pathway between said agent and said intermediary and a second data pathway between said intermediary and said agent, wherein at least one of said first and second data pathways is comprised of the Internet.

11. A method of remotely servicing a secure computer system, comprising:

ATTORNEY DOCKET NO. MCL-001XX
WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES, LLP
TEL. (617) 542-2290
FAX. (617) 451-0313

conveying a directive to be executed from a manager to an intermediary distinct from said secure computer system over a first secure communications pathway;

receiving, by an agent of said secure computer system, said directive to be executed from said intermediary over a second secure communications pathway; and

executing, by said agent, said directive to executed.

12. The method of claim 11, further comprising establishing said first secure communications pathway by said agent conveying intermediary-identifying data to said intermediary, and said intermediary conveying intermediary-identifying data to said manager.

13. The method of claim 11, further comprising establishing said first secure communications pathway by said manager conveying manager-identifying data to said intermediary.

14. The method of claim 11, further comprising establishing said second secure communications pathway by said agent conveying agent-identifying data to said intermediary.

15. The method of claim 11, further comprising establishing said second secure communications pathway by said intermediary conveying intermediary-identifying data to said agent.

16. The method of claim 11, wherein said establishment of said first secure communications pathway between said manager and said intermediary comprises using a web browser as said manager and accessing a web server as said intermediary.

17. The method of claim 11, further comprising:

returning, by said agent, results of an executed directive to said intermediary; and

returning, by said intermediary, said executed directive results to said manager.

18. A remote servicing system intermediary for a secure computer system, comprising:

a web server communicable, over a first secure communications link, with a manager running on a management computer system and, over a second secure communications link, with an agent running on said secure computer system,

said web server capable of spawning a first sub-process for receiving, over said first secure communications link, a directive to be executed by said secure computer system, and

said web server capable of spawning a second sub-process for transmitting, over said second secure communications link, said directive to be executed by said secure computer system.

19. The intermediary of claim 18, further comprising memory for storing a password received from said secure computer system for use in validating said manager.

20. The intermediary of claim 18, wherein said intermediary is operative to block pending receipt of said directive to be executed from said manager.

21. The intermediary of claim 18, further comprising a digital certificate to be conveyed to said agent and to said manager for authenticating said intermediary.

22. The intermediary of claim 18, further comprising an access database for defining which secure computer systems communicable with said intermediary said manager may define directives for.

23. The intermediary of claim 18, wherein said intermediary is operative to receive results of said executed directive from said secure computer system over said second secure communications link and is further operative to convey said results to said manager over said first secure communications link.

24. A method of operating an intermediary for the remote servicing of a secure computer system, comprising:

providing a first digital certificate to an agent running on said secure computer system in response to a request from said agent for the purpose of authenticating said intermediary to said agent;

storing identifying information received from said agent;

providing a second digital certificate to a manager in response to a request from said manager for the purpose of authenticating said intermediary to said manager;

receiving said identifying information from said manager for the purpose of authenticating said manager to said intermediary;

receiving, from said manager, a directive to be executed by said agent; and

forwarding said directive to said agent for execution.

25. The method of claim 24, further comprising:

receiving results of said executed directive from said agent; and

forwarding said results to said manager.

26. The method of claim 24, further comprising:

storing a list of secure computer systems communicable with said intermediary and for which said manager may define directives to be executed.

27. The method of claim 24, further comprising:

establishing a first secure communications pathway between said intermediary and said agent in response to said agent authenticating said intermediary based upon said provision of said first digital certificate from said intermediary to said agent.

28. The method of claim 24, further comprising:

establishing a second secure communications pathway between said intermediary and said manager in response to said manager authenticating said intermediary based upon said provision of said second digital certificate from said intermediary to said manager and in response said receipt of said identifying information from said manager by said intermediary.

ABSTRACT OF THE DISCLOSURE

The remote servicing of a secure computer system employs an intermediate network entity accessible to both a remote service provider and to an agent running on the secure computer system to be serviced. A service provider's computer runs a manager software module, while the system being serviced, or an agent on its behalf, runs an agent software module. An intermediary software module runs on the intermediate network entity. The mutually accessible intermediate network entity may be located outside firewalls protecting the system to be serviced or inside such firewalls though with the firewalls configured to allow selected access. Access to the intermediate network entity is limited by secure access protocols. After authentication, the manager submits to the intermediary one or more directives to be executed by the agent. The intermediary then sends the directives to the agent over a secure connection to the agent. The agent then executes the directive(s) upon receipt and sends the response to the intermediary via the secure connection. The manager is then capable of accessing the results via the intermediary.

223159

ATTORNEY DOCKET NO. MCL-001XX
WEINGARTEN, SCHURGIN,
GAGNEBIN & HAYES, LLP
TEL. (617) 542-2290
FAX. (617) 451-0313

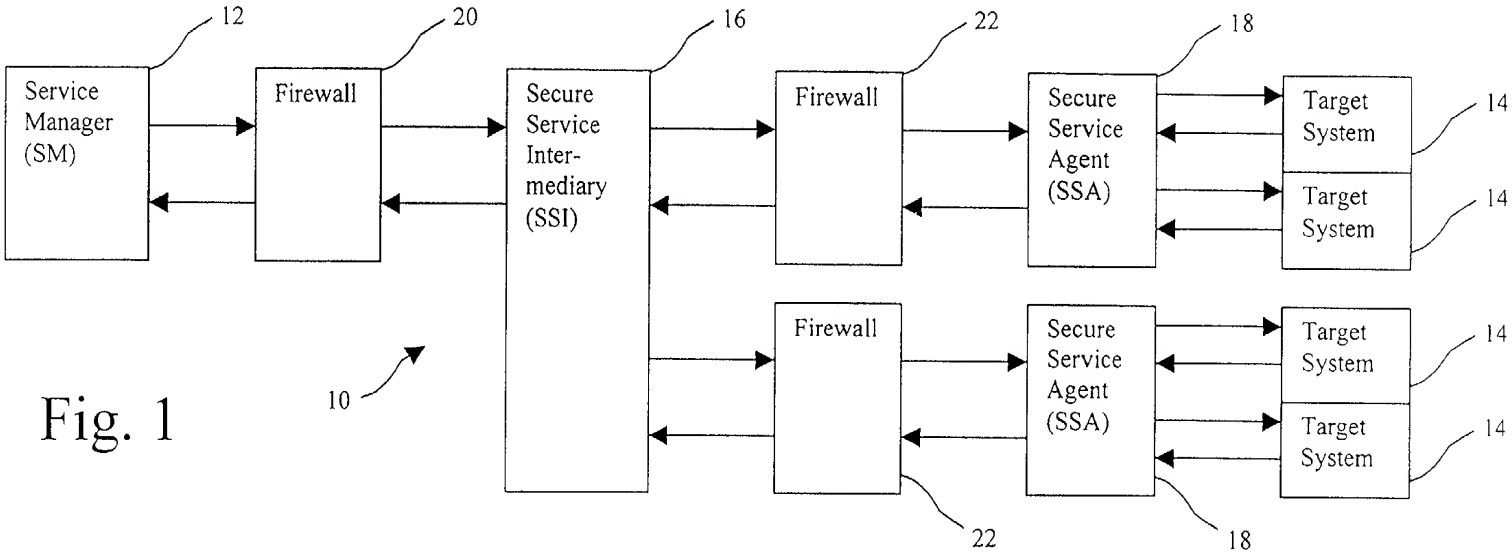

Fig. 1

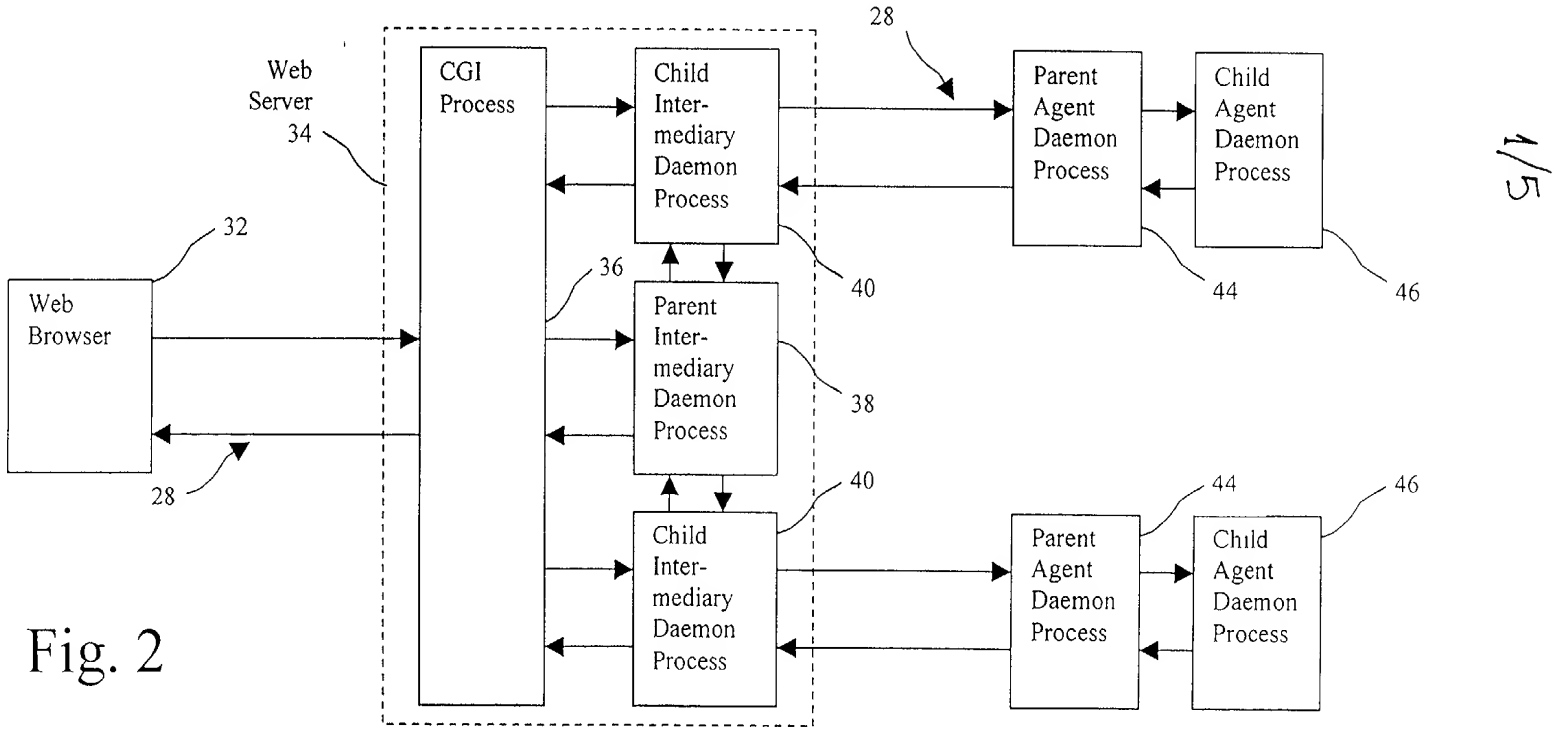

Fig. 2

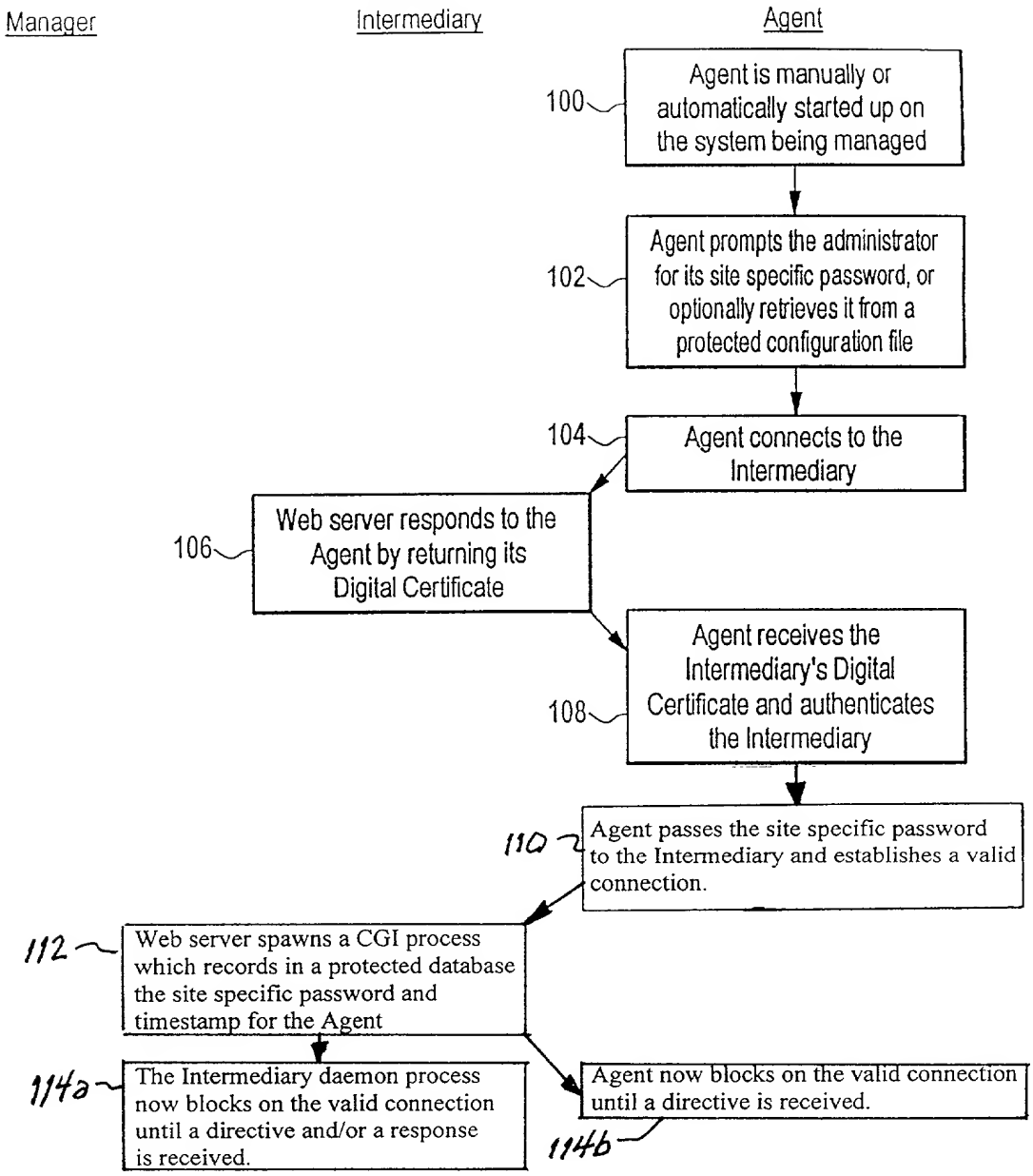
Manager
Intermediary
Agent
100
Agent is manually or automatically started up on the system being managed
102
Agent prompts the administrator for its site specific password, or optionally retrieves it from a protected configuration file
104
Agent connects to the Intermediary
106
Web server responds to the Agent by returning its Digital Certificate
108
Agent receives the Intermediary's Digital Certificate and authenticates the Intermediary
110
Agent passes the site specific password to the Intermediary and establishes a valid connection.
112
Web server spawns a CGI process which records in a protected database the site specific password and timestamp for the Agent
114a
The Intermediary daemon process now blocks on the valid connection until a directive and/or a response is received.
114b
Agent now blocks on the valid connection until a directive is received.

FIG. 3

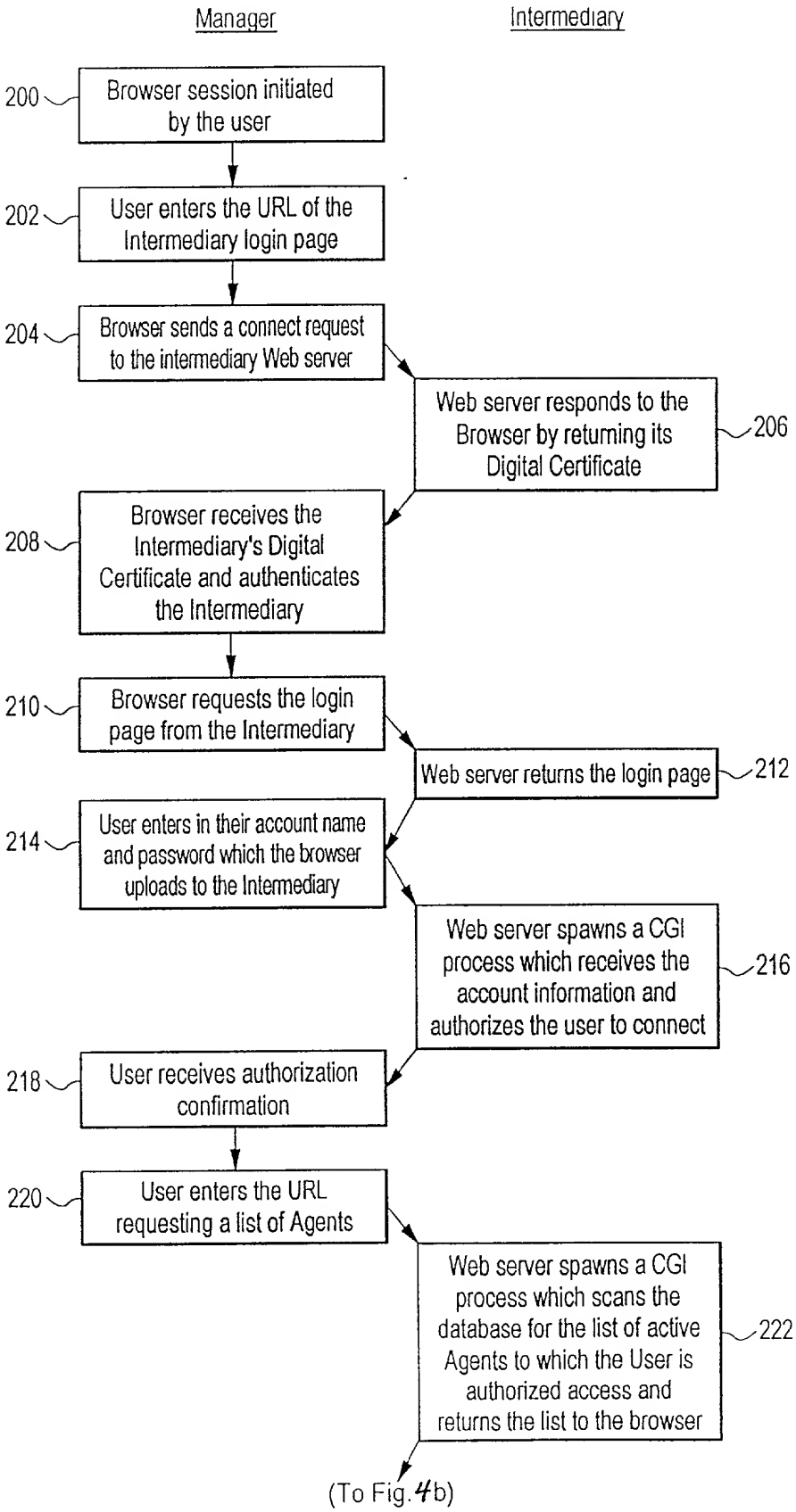
Manager
Intermediary
Agent
200
Browser session initiated by the user
202
User enters the URL of the Intermediary login page
204
Browser sends a connect request to the intermediary Web server
Web server responds to the Browser by returning its Digital Certificate
206
208
Browser receives the Intermediary's Digital Certificate and authenticates the Intermediary
210
Browser requests the login page from the Intermediary
Web server returns the login page
212
214
User enters in their account name and password which the browser uploads to the Intermediary
Web server spawns a CGI process which receives the account information and authorizes the user to connect
216
218
User receives authorization confirmation
220
User enters the URL requesting a list of Agents
Web server spawns a CGI process which scans the database for the list of active Agents to which the User is authorized access and returns the list to the browser
222
(To Fig. 4b)

FIG. 4a

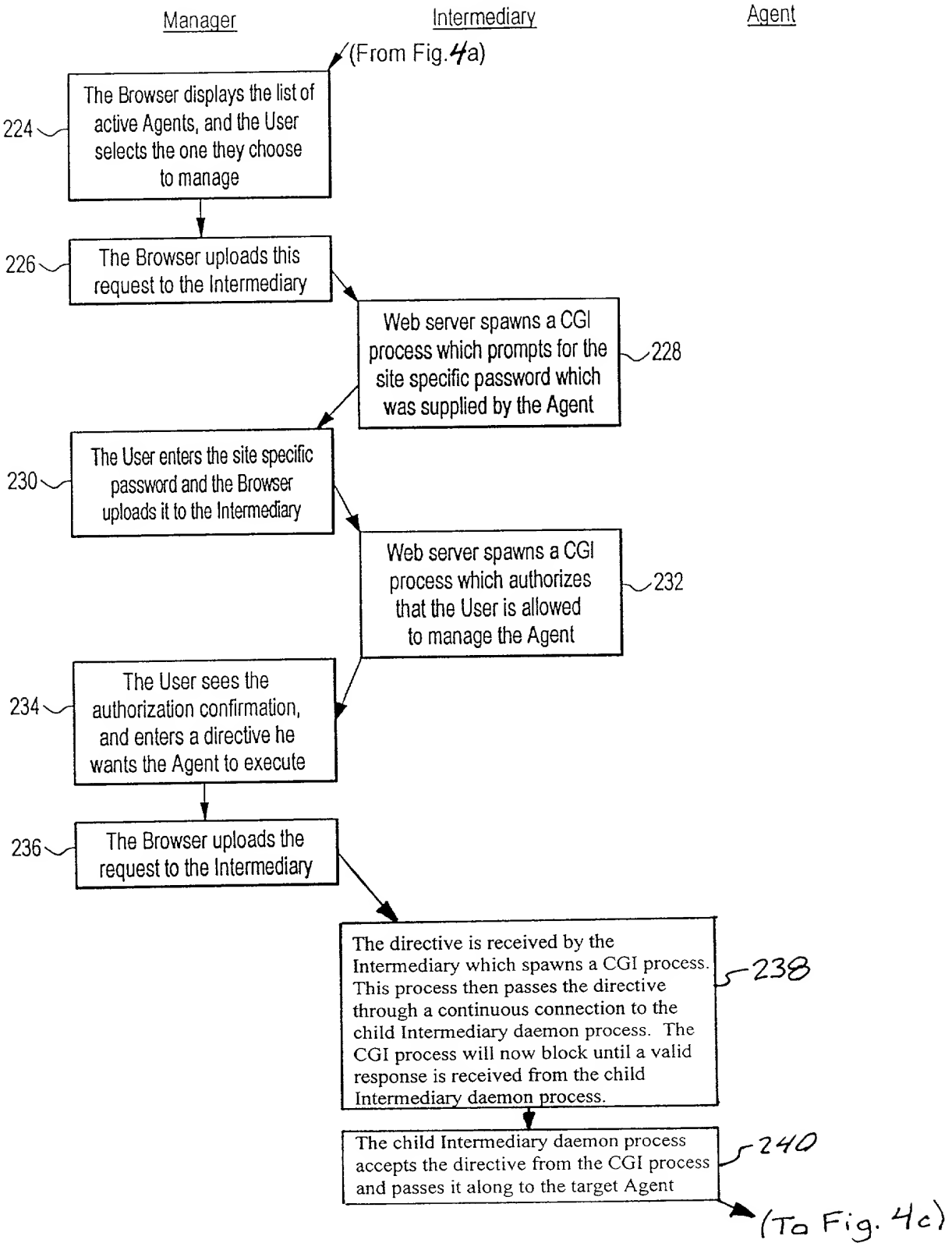
Manager
Intermediary
Agent
(From Fig. 4a)
The Browser displays the list of active Agents, and the User selects the one they choose to manage
224
The Browser uploads this request to the Intermediary
226
Web server spawns a CGI process which prompts for the site specific password which was supplied by the Agent
228
The User enters the site specific password and the Browser uploads it to the Intermediary
230
Web server spawns a CGI process which authorizes that the User is allowed to manage the Agent
232
The User sees the authorization confirmation, and enters a directive he wants the Agent to execute
234
The Browser uploads the request to the Intermediary
236
The directive is received by the Intermediary which spawns a CGI process. This process then passes the directive through a continuous connection to the child Intermediary daemon process. The CGI process will now block until a valid response is received from the child Intermediary daemon process.
238
The child Intermediary daemon process accepts the directive from the CGI process and passes it along to the target Agent
240
(To Fig. 4c)

FIG. 4b

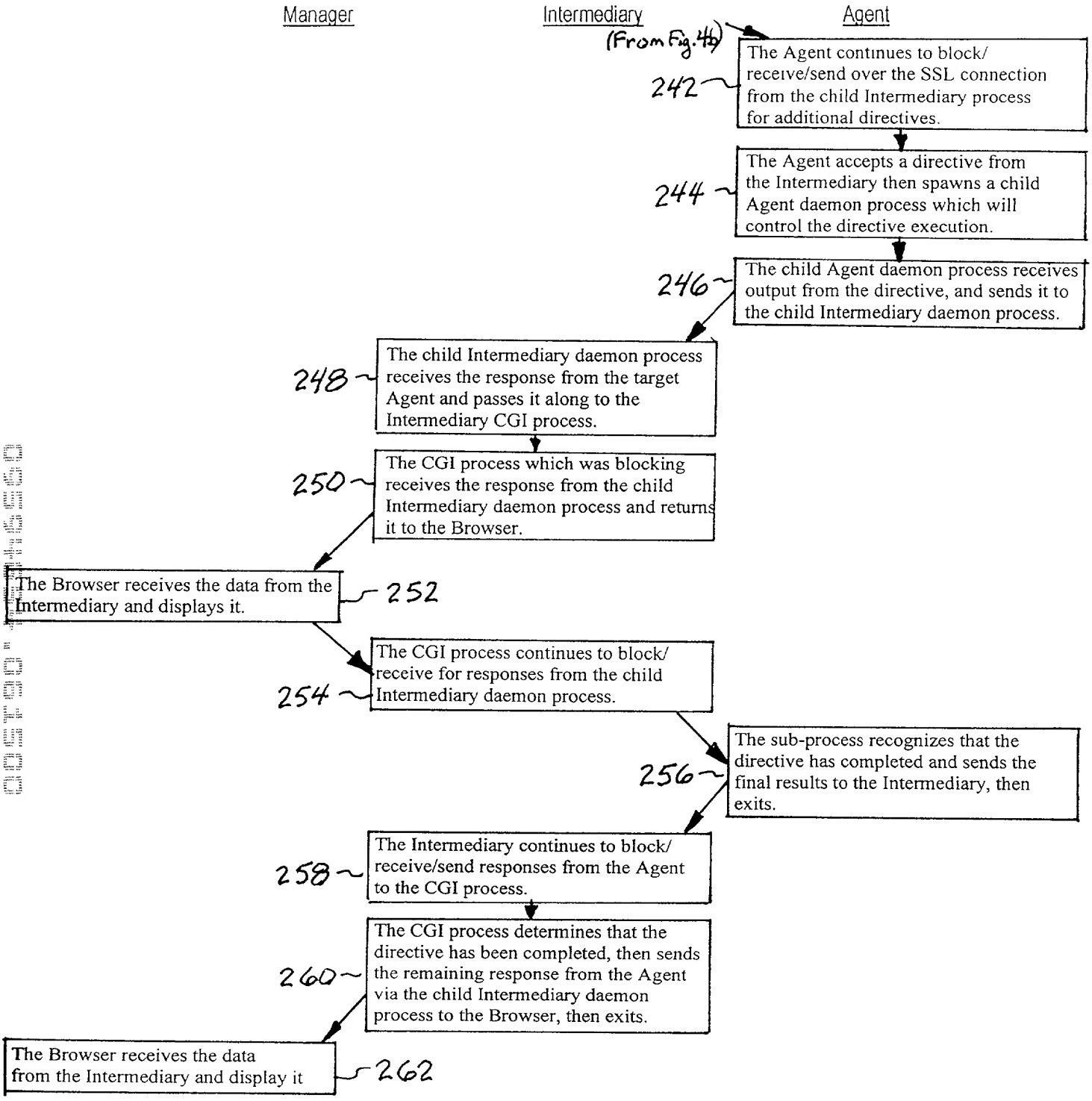
Manager
Intermediary
Agent
(From Fig. 4b)
The Agent continues to block/ receive/send over the SSL connection from the child Intermediary process for additional directives.
242
The Agent accepts a directive from the Intermediary then spawns a child Agent daemon process which will control the directive execution.
244
The child Agent daemon process receives output from the directive, and sends it to the child Intermediary daemon process.
246
The child Intermediary daemon process receives the response from the target Agent and passes it along to the Intermediary CGI process.
248
The CGI process which was blocking receives the response from the child Intermediary daemon process and returns it to the Browser.
250
The Browser receives the data from the Intermediary and displays it.
252
The CGI process continues to block/ receive for responses from the child Intermediary daemon process.
254
The sub-process recognizes that the directive has completed and sends the final results to the Intermediary, then exits.
256
The Intermediary continues to block/ receive/send responses from the Agent to the CGI process.
258
The CGI process determines that the directive has been completed, then sends the remaining response from the Agent via the child Intermediary daemon process to the Browser, then exits.
260
The Browser receives the data from the Intermediary and display it
262

FIG. 4c

DECLARATION AND POWER OF ATTORNEY

As a below-named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name;

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled: SECURE REMOTE SERVICING OF A COMPUTER SYSTEM OVER A COMPUTER NETWORK

the specification of which (check one):

[X] is attached hereto. [ ] was filed _____ as Application No._____
amended on _____(if applicable).

[ ] was filed as PCT International Application No. _____ on _____,
and was amended under PCT Article 19 on _____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations §1.56(a).

I hereby claim foreign priority benefits under Title 35, USC §119(a)-(d) of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)		Date Filed	Priority Claimed	
(Number)	(Country)	(Day/Month/Year)	[ ] Yes	[ ] No
(Number)	(Country)	(Day/Month/Year)	[ ] Yes	[ ] No

I hereby claim the benefit under Title 35, USC §119(e) of any United States provisional application(s) listed below:

60/160,985 (Application Number)	October 22, 1999 (Filing Date)
(Application Number)	(Filing Date)
(Application Number)	(Filing Date)

Express Mail No.

EL418426925US

I hereby claim the benefit under Title 35 USC §120 of any United States application(s) listed below and insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35 USC §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

(Application No.)	(Filing Date)	(Patented/pending/abandoned)
(Application No.)	(Filing Date)	(Patented/pending/abandoned)

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) to prosecute this application and transact all business connected therewith in the Patent and Trademark Office, and to file with the USRO any International Application based thereon.

Stanley M. Schurgin, Reg. No. 20,979
Charles L. Gagnebin III, Reg. No. 25,467
Paul J. Hayes, Reg. No. 28,307
Victor B. Lebovici, Reg. No. 30,864
Eugene A. Feher, Reg. No. 33,171
Beverly E. Hjorth, Reg. No. 32,033
Holliday C. Heine, Reg. No. 34,346
Gordon R. Moriarty, Reg. No. 38,973

Address all correspondence to:

WEINGARTEN, SCHURGIN, GAGNEBIN & HAYES LLP
Ten Post Office Square
Boston, Massachusetts 02109
Telephone: (617) 542-2290
Telecopier: (617) 451-0313

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole Inventor: Brian Stevens		
City of Residence: New Boston	State or Country: New Hampshire	Country of Citizenship: United States
Post Office Address: 165 Mont Vernon Road	City: New Boston	State or Country: New Hampshire — Zip Code: 03070
Signature: (Please sign and date in permanent ink.) X Brian Stevens		Date signed: X 6/7/00

213554